Using Lucene for Developing a Question-Answering Agent in Portuguese

Hugo Gonçalo Oliveira
CISUC, Department of Informatics Engineering, University of Coimbra, Portugal
https://eden.dei.uc.pt/~hroliv/
hroliv@dei.uc.pt

Ricardo Filipe
ISEC, Polytechnic Institute of Coimbra, Portugal
ricardo.ferreira.filipe@gmail.com

Ricardo Rodrigues
CISUC, University of Coimbra, Portugal
ESEC, Polytechnic Institute of Coimbra, Portugal
rmanuel@dei.uc.pt

Ana Alves
CISUC, University of Coimbra, Portugal
ISEC, Polytechnic Institute of Coimbra, Portugal
ana@dei.uc.pt

Abstract

Given the limitations of available platforms for creating conversational agents, and that a question-answering agent suffices in many scenarios, we take advantage of the Information Retrieval library Lucene for developing such an agent for Portuguese. The solution described answers natural language questions based on an indexed list of FAQs. Its adaptation to different domains is a matter of changing the underlying list. Different configurations of this solution, mostly on the language analysis level, resulted in different search strategies, which were tested for answering questions about the economic activity in Portugal. In addition to comparing the different search strategies, we concluded that, towards better answers, it is fruitful to combine the results of different strategies with a voting method.

2012 ACM Subject Classification Information systems → Search interfaces; Information systems → Question answering; Computing methodologies → Natural language processing

Keywords and phrases information retrieval, question answering, natural language interface, natural language processing, natural language understanding

Digital Object Identifier 10.4230/OASIcs.SLATE.2019.2

Funding This work was partially funded by FCT's INCoDe 2030 initiative, in the scope of the demonstration project AIA, "Apoio Inteligente a Empreendedores (*chatbots*)".

1 Introduction

A natural way of interacting with computational systems is by communicating with them in the same way we communicate with other humans: using natural language. This is indeed one of the long-term goals of Natural Language Processing (NLP), currently materialised in Intelligent Personal Assistants, like Apple's Siri or Amazon's Alexa, which accept commands in natural language, and are running on our smartphones, smartwatches or smart homes. In fact, since ELIZA [22], chatbots and dialog systems have become more human-like, able to engage in informal conversations and to learn from user input.

This shift is so present that most organisations are adopting one or more assistants of this kind as alternative channels to interact with their customers. This high demand lead

8th Symposium on Languages, Applications and Technologies (SLATE 2019).
Editors: Ricardo Rodrigues, Jan Janoušek, Luís Ferreira, Luísa Coheur, Fernando Batista, and Hugo Gonçalo Oliveira; Article No. 2; pp. 2:1–2:14
OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

to the development of several platforms for easing the creation of conversational agents [3], through a high-level interface, which reduces much of the time that would, otherwise, be spent on the process. Yet, their pipeline is often not so much customisable on the lower-level pre-processing, especially for non-English languages. Furthermore, some of those platforms are proprietary and have usage restrictions. This means that, from the beginning, solutions based on them are tied to the underlying pre-processing and the so-called Natural Language Understanding (NLU) mechanisms. Among other drawbacks, this prevents their experimentation with different techniques or language resources.

Still, in many scenarios, dialog capabilities are not necessary, and an improved search mechanism, possibly deployed as a question-answering (QA) agent, is enough. Such an agent would have the main goal of: (1) processing user input; (2) matching it with questions from its knowledge-base; (3) providing adequate answers. A knowledge-base for such an agent should cover questions about the target organisation and the services it provides, with a focus on those frequently asked by customers. This suggests that such a knowledge base could be made of frequently asked questions (FAQs).

This paper describes how a QA agent can be developed on top of Lucene, an open-source solution for Information Retrieval (IR), which provides high-performance full text indexing and searching, while offering a high level of customisation, namely concerning the applied pre-processing, indexed information and search metrics. Since our main domain of application is on Portuguese text, the previous reasons lead us to tune the agent for Portuguese, with the integration of specific pre-processing for this language. The developed system indexes a list of FAQs and tries to match user input, written in natural language, with the available questions. Once a question is matched, its answer is retrieved. Furthermore, as different metrics can be applied for matching, the developed agent enables the integration of different strategies for this purpose, considering different search fields, analysis or levels of tolerance. Ultimately, all the available metrics can be used in parallel, towards a better decision on the best candidate answer.

In the remainder of this paper, we provide a brief overview on the background scientific areas that support this work, covering search technologies, chatbots and dialog systems, and also their intersection. After that, we describe the architecture of our QA agent, including details on tuning Lucene for our purposes, tools and resources exploited, and also on the implemented search strategies. We admittedly tried to balance the previous description between the scientific contributions and implementation details, having in mind future applications of this agent, possibly by other researchers, but also those only interested in tuning Lucene for Portuguese. Before concluding, we report on the utilisation of the QA agent in a specific scenario. For demonstration purposes, it was tested with a list of FAQs obtained from the Portuguese Entrepreneur's Desk (in Portuguese, "*Balcão do Empreendedor*") and different search strategies were used for answering variations of the questions in that list. Besides comparing the implemented search strategies, we concluded that answer accuracy is higher when results of more than one strategy are combined.

2 Background and Related Work

Since ELIZA, chatbots and dialog systems have become more human-like, capable of engaging in informal conversations (see, e.g., Mitsuku[1] or Rose[2]), and of learning from human input (see,

[1] https://www.pandorabots.com/mitsuku/

[2] http://brilligunderstanding.com/rosedemo.html

e.g., Cleverbot[3]). Common approaches for developing such a system exploit large collections of text, often including conversations. Generative systems try to model conversations with a neural network that learns to decode a sequence of text and translate it to another sequence, used as a response [20]. They are generally scalable, versatile, and always generate a response, but have limitations when it comes to performing specific tasks. They make few assumptions about the domain and generally have no access to external sources of knowledge, which means that they can rarely handle factual content. They also tend to be too repetitive and provide inconsistent, trivial or meaningless answers.

Task-oriented agents tend to follow other strategies and integrate Information Retrieval (IR) and Question Answering (QA) techniques, in order to find the most relevant response to requests in natural language. The long-established task of IR has the goal of finding information automatically in a collection of documents, typically a large one. The input of a traditional IR system is a query that represents an information need, typically in the form of keywords, to be answered with a list of documents. There are countless models for retrieving relevant documents for a query and for ranking them.

Yet, a traditional IR system does not interpret the meaning of the query. Relevant documents are generally selected because they mention the keywords, possibly their stems, or are about the topics they convey. Diversely, automatic QA [9, 21] has the main goal of finding answers to questions formulated in natural language. Answers can be retrieved from a knowledge base [16] or from a collection of documents [12]. This has similarities to IR, especially the latter [12], but queries have to be further interpreted, possibly reasoned – this is where NLU capabilities may be necessary – , while answers are expected to go beyond just a list of documents.

Given a user input, IR-based conversational agents search for the most similar request on the corpus and output its response (see, e.g., [10]). They rely on an IR system for indexing the documents of the corpus and, in order to identify similar texts and computing their relevance, they apply IR ranking techniques, often based on the vector space model and on the cosine between the request vector and vectors of indexed requests or possible responses. But those measures can also be combined with an alternative ranking function learned specifically for that purpose. This can be achieved, for instance, with a regression model that considers several lexical or semantic features to measure semantic textual similarity [5].

As opposed to the generative approach, IR-based conversational agents do not handle very well requests for which there is no similar text in the corpus. Nevertheless, an alternative IR-based strategy can still be followed, in this case, for finding similar texts in a more general corpus, such as movie subtitles [14]. In order to minimise the limitations of generative and IR-based approaches for chatbots, some authors tried to combine them. For instance, IR techniques have been used for reordering a set of generated answers [19].

The high demand for chatbots and virtual assistants lead to the development of several platforms [3] – for instance, DialogFlow[4], Wit.ai[5], Luis[6], Watson Assistant[7] – for easing the creation of such systems, through high-level interfaces. This turns out to be an easy solution or creating conversational agents out-of-the-box and reduces much of the time that would, otherwise, be spent on this process. However, not all of such platforms are completely free – some are proprietary and their utilisation may be dependent on, or restricted according to, a paid license – and most are not as flexible as a NLP researcher would wish, especially when

[3] https://www.cleverbot.com/
[4] https://dialogflow.com/
[5] https://wit.ai/
[6] https://luis.ai/home
[7] https://www.ibm.com/cloud/watson-assistant/

it comes to non-English languages.

Those platforms are usually based on the concepts of intent – i.e., purpose of the users input – and entity – i.e., terms that are relevant for the intent [3]. For instance, in the question *What is the weather like in Coimbra?*, the intent would be to know weather information, while Coimbra is the target entity. We analysed some platforms and made some experiments with DialogFlow. Yet, version 1 of its SDK did no enable intent management. This had to be done through the web interface in the platform website. Version 2 beta has had some improvements, but, still, does not enable to control the NLU techniques applied, for example, how user input is matched with intents. This means that it does not enable, for instance, the experimentation of different models for this purpose – e.g., using language-specific tools – nor the integration of a lexical resource – e.g., for handling synonyms or related words.

In some scenarios, dialog capabilities are not that crucial, and a QA agent, capable of matching natural language requests with known questions is enough. In this case, when available, lists of frequently asked questions, typically abbreviated to FAQs, are valuable resources for exploitation, due to their nature and structure. In fact, QA systems have exploited FAQs on the web for open-domain QA [11]; in the last decade, there has been interest on SMS-based interfaces for FAQs [13]; and there has also been a shared task on QA from FAQs in Italian [4]. The previous QA system relied on Lucene for indexing and as a baseline for retrieving documents [4]. Furthermore, FAQ-based QA agents often pre-process text in questions, answers and user requests, applying tokenisation and stopword removal operations. When matching user requests with FAQs, they exploit word overlap or the presence of similar words. In some cases, synonyms [13, 15], acronyms [7] or distributional semantic features [7] are also considered.

3 System Architecture

As stated earlier, this paper is about the development of a flexible QA agent for Portuguese, based on a list of FAQs that might be changed, depending on the agent's purpose. As other IR-based QA [11, 13] or conversational agents [14, 5], it was built on top of Apache Lucene[8], an open-source Java library for IR, which provides high performance full-text indexing and search capabilities. Specifically, we used version 7.7.0 of this library.

Lucene enables the representation of textual documents in a set of fields, some containing single values and other longer texts. Not all fields have to be indexed, specifically those that are not considered in searches. In its basic use case, Lucene would represent each document by a text field with the textual contents of the document. This enables full-text search on the collection of documents. Additional fields may be useful for adding metadata, such as the document location, and they can also be used for storing and indexing the result of the text after some pre-processing operations.

The diagram in figure 1 shows the question-answering flow in our agent. Textual input by the user is interpreted as a question and then analysed, to be prepared for the retrieval phase. The analysis made here should be equivalent to the one made in the creation of the index (see section 3.1). It may cover pre-processing tasks such as tokenisation or synonym expansion. After the analysis, the input is matched with a suitable question on the knowledge base. In fact, Lucene will retrieve a list of candidate documents (questions), ranked according to their relevance. The agent will retrieve not only the top-n questions, but also their answers, which are then shown to the user. Hopefully, one of them will be the right answer.

[8] `https://lucene.apache.org/`

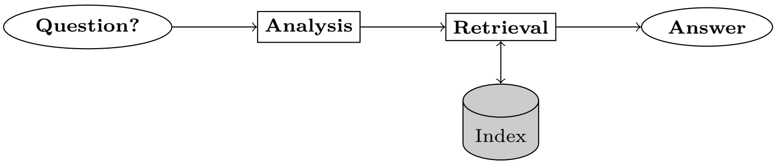

Figure 1 Question-answering flow.

The remainder of this section details how Lucene is adjusted in the implementation of the aforementioned flow. It starts with the creation of the index, moves on to a brief description of external tools and resources exploited in the analysis, and finally enumerates some of the strategies applied for searching on the index.

3.1 Index creation

The starting point for creating the underlying index for the QA agent is a list of questions and their answers, both written in natural language. This can be a list of FAQs, such as those commonly found in institutional websites, which include answers to questions frequently asked about the institution, in order to help (potential) customers finding out more about it, as fast as possible, and, at the same time, avoiding a congestion of contacts through other channels, such as e-mail or telephone.

The QA agent may work with any list of FAQs, as long as, before indexing, this list is organised in a text file with questions and answers interleaved, in lines respectively starting with `P:` and `R:`. This means that, changing the FAQs and creating a new index is all it takes for adapting the agent to a different domain. Due to our goal and performed adaptations, besides the input format, the only restriction is that FAQs are written in Portuguese.

When creating an index, the text in the FAQs file is analysed by Lucene. This is performed by an instance of the Analyzer class of Lucene's API, which has the main purpose of tokenising the text. Lucene offers several analysers out-of-the-box, including StandardAnalyzer and PortugueseAnalyzer. The former applies standard tokenisation rules, converts text to lower case and ignores English stopwords. PortugueseAnalyzer, available under the package `org.apache.lucene.analysis.pt`, extends StandardAnalyzer but, as its name suggests, considers some specificities of Portuguese: it uses a list of Portuguese stopwords instead, borrowed from Snowball[9], and applies rules for stemming Portuguese words and stripping off accents, with PortugueseLightStemmer.

We implemented search strategies using each of the previous analysers, but, for higher control, we also reimplemented the Analyzer class and created CustomPortugueseAnalyzer. This analyzer has the option of stemming or not, and of considering a map for words and their possible synonyms. The latter is easily integrated in the Analyzer with an instance of the SynonymMap class, under the package `org.apache.lucene.analysis.synonym`. Search recall should benefit from this map, as it helps dealing with language variation. Producing lemmatised versions of each question-answer pair and indexing them in specific fields should also have a positive impact on handling language variation. Lemmatisation resorts to an external tool, further introduced in section 3.2.

Figure 2 illustrates the indexing procedure, to be run for each list of FAQs. Once the index is ready, it will only have to be created again if there are changes on the data (e.g., a new question is added) or a different analysis is required. In the index, each question is

[9] `http://snowball.tartarus.org/algorithms/portuguese/stop.txt`

represented as illustrated in table 1, namely with four fields: original question (P), lemmatised question (PL), answer (R), and lemmatised answer (RL). All of these fields are searchable, though no implemented search strategy considers the four of them.

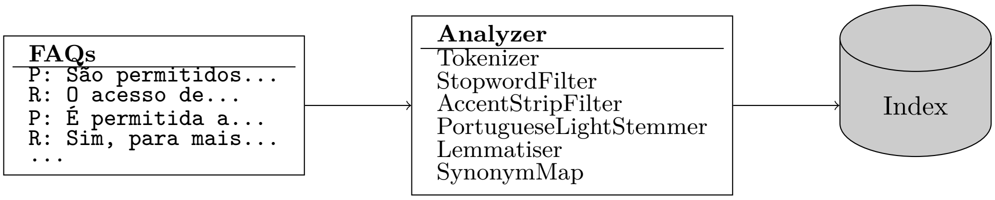

Figure 2 Indexing the list of FAQs.

Table 1 Document fields in Lucene and their contents for one question.

Field	Content
P	`São permitidos animais em estabelecimentos de restauração ou bebidas?` *(Are pets allowed in restaurants or bars?)*
PL	`Ser permitido animal em estabelecimento de restauração ou bebida?` *(Be pet allow in restaurant or bar?)*
R	`O acesso de animais é permitido apenas às esplanadas, exceto cães de assistência que podem aceder a toda a área frequentada pelos clientes...` *(Animal access is allowed only to the terraces, except for service dogs that can access the entire area used by customers.)*
RL	`O acesso de animal ser permitido apenas à esplanada, exceto cão de assistência que pode aceder a toda a área frequentada pelo cliente...` *(Animal access be allow only to the terrace, except for service dog that can access the entire area use by customer.)*

3.2 External Tools & Resources

As mentioned in the previous section, PortugueseAnalyzer has several useful filters for removing Portuguese stopwords, stripping off accents and stemming. Yet, we created a specific instance of the Analyzer class, which includes some of the previous filters but is further augmented, to be used by some search strategies. First, questions and answers were lemmatised as an alternative to stemming. Although both stemming and lemmatisation are normalisation alternatives, they are applied at different stages. Stemming is applied by Lucene's Analyzer as part of the indexing and retrieval process, while the lemmatised versions of the question and of the answer are stored and indexed in specific fields and applied to every query, when these fields are considered. In opposition to stemming, lemmatisation always results in a valid word (a lexeme), generally the dictionary entry for the original word. For this purpose, lemmatisation depends on a previous morphology analysis and thus requires more language-specific knowledge. Contrarily, stemming simply cuts off the end of the words, and sometimes also the prefixes, which is not always enough to match words in different forms. Consider, for instance, the words *question*, *questions*, *mouse*, *mice*, *is*, and *were*. Possible stems are *quest*, *quest*, *mous*, *mic*, *is*, and *wer*, while their lemmas are *question*, *question*, *mouse*, *mouse*, *be*, and *be*.

Lemmatisation was performed with LemPORT, part of the NLPPort [18] toolkit. To enable lemmatisation, both question and answer had to be first tokenised and part-of-speech tagged with tools that are also part of the previous toolkit, namely TokPort and TagPort.

The second adaptation took advantage of Lucene's SynonymMap class, which, although easy to integrate in an Analyzer, was not available for Portuguese. As its name suggests, an instance of a SynonymMap maps words with their possible synonyms. For our agent, the source of this map was a lexical knowledge base extracted from ten different Portuguese lexical resources [8], freely available online[10]. This knowledge base is represented in a text file of which we show some lines in figure 3. Each line contains relations of different types, between two words, each followed by a number that represents the number of resources where the relation was found. Since this number is related to reliability and acceptability of the semantic relation, we decided to use only the 3,483 synonym pairs with a number of resources higher than five – i.e., the pair occurs in at least six resources. Relations with a lower number or of other types were not used.

Figure 4 shows the map resulting from the relations in figure 3. Depending on the relations in the list, a word can be mapped to one or more synonyms.

```
multa SINONIMO_N_DE coima 6
procedimento SINONIMO_N_DE comportamento 6
permitir SINONIMO_V_DE admitir 6
permitir SINONIMO_V_DE deixar 6
autorizar SINONIMO_V_DE permitir 6
consentir SINONIMO_V_DE permitir 7
permitir SINONIMO_V_DE conceder 7
```

Figure 3 Selected lines of the lexical knowledge base file.

```
       multa   →  coima
       coima   →  multa
     conduta   →  procedimento
procedimento   →  conduta, comportamento
comportamento  →  procedimento
    permitir   →  admitir, deixar, autorizar, consentir, conceder
     admitir   →  permitir
      deixar   →  permitir
   autorizar   →  permitir
   consentir   →  permitir
    conceder   →  permitir
```

Figure 4 Synonym Map resulting from the synonym pairs in Figure 3.

3.3 Search strategies

Even with the performed analysis and the resulting index, some issues arise about the best way to exploit all the strategies towards the best results for a user query. For instance, in some cases, matching the input text with the original questions might be more fruitful than lemmatising both; answers often contain relevant information for this process; not to mention that the input text might contain typos or spelling mistakes, so tolerance is sometimes necessary. In order to cover all the aforementioned issues, different search strategies can be implemented, considering the contents of different fields, applying fuzzy instead of exact search, or using different similarity metrics. We ended up not exploring the latter. All the implemented search strategies rely on the BM25 similarity [17], a probabilistic model for IR, currently the default ranking method of Lucene, and an alternative to the classic TF-IDF.

Alternatively, we implemented strategies that match the input with different indexed fields (Question, Question and Answer, or their lemmatised versions), and based on different analysers (Standard, Portuguese, Custom), considering synonyms or not. Table 2 lists all

[10] http://ontopt.dei.uc.pt/index.php?sec=download_outros

the implemented search strategies, together with their configuration. It should be noted that some of the analysis made to the same fields is incompatible (e.g., StandardAnalyzer vs PortugueseAnalyzer, considering synonyms vs not). To cope with this situation, different indexes are created and not all search strategies share the same index.

Table 2 Search strategies and their configuration.

Strategy	Fields	Analyzer	Normalisation	Synonyms
VanillaQuestion	P	StandardAnalyzer	None	No
VanillaQuestionAnswer	P, R	StandardAnalyzer	None	No
StemQuestion	P	PortugueseAnalyzer	Stemming	No
StemQuestionAnswer	P, R	PortugueseAnalyzer	Stemming	No
StemSynsQuestion	P	CustomPTAnalyzer	Stemming	Yes
StemSynsQuestionAnswer	P, R	CustomPTAnalyzer	Stemming	Yes
LemmaQuestion	PL	CustomPTAnalyzer	Lemmatisation	No
LemmaQuestionAnswer	PL, RL	CustomPTAnalyzer	Lemmatisation	No
LemmaSynsQuestion	PL	CustomPTAnalyzer	Lemmatisation	Yes
LemmaSynsQuestionAnswer	PL, RL	CustomPTAnalyzer	Lemmatisation	Yes

We also created a fuzzy version of each strategy, taking advantage of Lucene's fuzzy search feature, which allows searching for words with an edit distance of at most two characters. This also increases tolerance to misspelled words or other typos. Implementing the fuzzy versions was a matter of adding the ~ (tilde) search operator to the end of each term in the query.

4 Case study

In order to test the QA agent in a real scenario, it was used for indexing a list of FAQs in Portuguese, thus allowing user queries that would be matched against the available questions, for which the answers would then be presented. This instantiation of the QA agent confirmed that adapting the system to a domain is just a matter of providing a list of FAQs in the desired format, while a list of acronyms can optionally be exploited. This section is about experiments and results in the previous domain. It starts by describing the list of FAQs, the list of acronyms, and a manually created evaluation dataset. After that, results on the previous dataset are presented and discussed for different search strategies.

4.1 Domain description

The FAQs used for this instantiation of the QA agent were collected from the "*Balcão do Empreendedor* (BDE)" portal, the Portuguese Entrepreneur's Desk[11], which is a single point of access to digital services related to the exercise of economic activity in Portugal. BDE is directed to entrepreneurs who wish to perform services and obtain information inherent to the economic activities that they practice. Specifically, the list used for this purpose included 120 FAQs from the Guide for the Application of the RJACSR ("*Guia de Aplicação do Regime Jurídico de Acesso e Exercício de Atividades de Comércio, Serviços e Restauração*") and 56 from the Legislation of the Local Accomodation ("*Legislação do Alojamento Local*")[12], thus totalling 176 FAQs.

[11] https://bde.portaldocidadao.pt/evo/balcaodoempreendedor.aspx (retrieved on June 2018)
[12] Both of these documents were downloaded from BDE on June 2018.

In order to increase recall, while indexing the aforementioned FAQs, Lucene's SynonymMap, used by four of the search strategies, was enriched with 38 acronyms and their full meaning. The aforementioned list was provided by the Portuguese Administrative Modernisation Agency (AMA) and included acronyms commonly used in the BDE, such as RJACSR, AL or MB, respectively for "*Balcão do Empreendedor*", "*Regime Jurídico das Atividades de Comércio, Serviços e Restauração*" (Legal Regime for Trade, Services and Catering Activities), "*Alojamento Local*" (Local Accommodation), or "*Multibanco*" (ATM).

These are the only adaptations required for using the developed QA agent on a specific domain. In fact, the list of acronyms is optional and not used by the majority of the search strategies implemented.

4.2 Evaluation dataset

In order to test how well the agent was doing its job, a more systematic evaluation was designed. Bearing in mind that, in most cases, users will not search for the exact question, variations of the original questions were also created and then used for assessing the system. Those variations included paraphrases or closely-related questions, including some in which words were on a different case or accents were missing. They were produced manually by three native Portuguese-speaking volunteers, who also assigned them to a correct answer in the list of FAQs. The resulting list had a total of 447 different questions, which means that, on average, it had ≈ 2.5 ways of asking each original question. It should be noted that not all of the original questions had the same number of variations produced for. Table 3 shows examples of the variations produced, in lines starting with "*P". Lines starting with a "P" and an "R" mark the original question and its answer, respectively.

4.3 Performance of search strategies

Each implemented search strategy was used to answer each question in the evaluation dataset. In this case, we considered that the answer would be the highest ranked search result, corresponding to one of the FAQs and its answer. Once an answer was selected, we checked whether it was the same of the original question (correct) or not (incorrect). In the end, we computed the accuracy of the QA agent, given by the ratio of correct answers to all question variations on the dataset (excluding the original questions).

Table 4 shows the results obtained for each search strategy. Its figures confirm that, due to language variation, matching questions in natural language is a challenging task. The best strategy, with 70% of the questions answered correctly, is the fuzzy version of VanillaQuestionAnswer. This is surprising, because this strategy relies on Lucene's StandardAnalyzer, which makes minimal pre-processing and has no knowledge specific of the Portuguese language. On the other hand, this is a fuzzy search, where variations of two characters per word in the query are accepted. This suggests that adding tolerance with fuzzy search might have a similar or even more positive effect than ignoring stopwords and normalising text with stemming or lemmatisation operations. Yet, fuzzy search only improves accuracy when it is applied to the vanilla search strategies. For strategies that include text normalisation, fuzzy searches always lead to accuracies lower than the normal search.

Accuracy is always better when the agent searches on both question and answer, which suggests that both should be exploited in this process. Finally, considering synonyms (and acronyms) increases the accuracy, but only when lemmas are used. This makes sense because the synonym pairs are established between lemmas, not inflected words nor stems.

Table 3 Original questions, answers, and manually created variations, for testing purposes.

P	*São permitidos animais em estabelecimentos de restauração ou bebidas?*
***P**	*podem entrar animais num estabelecimento abrangido pelo rjacsr*
***P**	*Que animais podem entrar num bar ou restaurante?*
R	*O acesso de animais é permitido apenas às esplanadas, exceto cães de assistência que podem aceder a toda a área frequentada pelos clientes...*
P	*Qual a coima aplicável às contraordenações graves?*
***P**	*coima para contraordenação grave*
***P**	*sanção das infrações graves*
***P**	*Qual o valor da multa para contraordenações graves?*
R	*As contraordenações graves são sancionáveis com coima: a) Tratando-se de pessoa singular, de € 1 200,00 a € 3 000,00;b) Tratando-se de microempresa, ...*
P	*Se a exploração do meu apartamento passar para outra pessoa, tenho de fazer um novo registo?*
***P**	*o meu imovel de alojamento local passou para outra pessoa, o que preciso de fazer*
***P**	*É necessário renovar o registo de um apartamento de que deixei de explorar?*
R	*Não, mantendo-se o mesmo estabelecimento de alojamento local, apenas é necessário efetuar uma alteração ao respetivo registo, através do balcão único eletrónico, ...*
P	*No alojamento local é obrigatória a certificação energética? Em que termos deve ser efetuada?*
***P**	*estabelecimento de alojamento local deve ter certificação energética*
***P**	*certificação energética do alojamento local*
***P**	*Como fazer a certificação energética do meu alojamento local?*
***P**	*Qual o procedimento para certificar energeticamente o meu alojamento local?*
R	*De acordo com esclarecimento da DGEG (Direção-Geral de Energia e Geologia) se os estabelecimentos de alojamento local se reportarem a edifícios ou frações autónomas ...*

Table 4 Accuracy of different search strategies.

Strategy	**Correct answers**			
	Normal		**Fuzzy**	
VanillaQuestion	275	(61%)	292	(65%)
VanillaQuestionAnswer	296	(66%)	314	(70%)
StemQuestion	297	(66%)	258	(57%)
StemQuestionAnswer	309	(69%)	263	(58%)
StemSynsQuestion	290	(64%)	244	(54%)
StemSynsQuestionAnswer	308	(68%)	237	(53%)
LemmaQuestion	279	(62%)	264	(59%)
LemmaQuestionAnswer	296	(66%)	292	(65%)
LemmaSynsQuestion	270	(60%)	252	(56%)
LemmaSynsQuestionAnswer	302	(67%)	280	(62%)

4.4 Combining Search Strategies

Implemented search strategies are substantially different and some seem to have a complementary nature, which leads to the selection of different answers. This made us wonder whether their search results could be combined, all contributing to the selection of better answers.

To test our hypothesis, we adopted a consensus-based voting method, BordaCount [6], which considers that searches often retrieve more than one result, in a ranked list. BordaCount scores candidates according to their rank on several lists. The higher the rank, the higher the score. More precisely, the score of the first candidate on a rank will be equal to the number of considered positions. For instance, if we consider the top-5 candidates, the first candidate gets 5 points and the fifth gets 1 point. The selected answer will be the one of the FAQ that has the highest final score, obtained by summing all of its partial scores, in all the considered lists, in this case, retrieved by each search strategy.

Tables 5 and 6 illustrate how this method works with three search strategies (VanillaQuestion, StemQuestionAnswer, LemmaSynsQuestionAnswer), for the query "*o que acontece se perto de uma sexshop for construido um espaço para crianças*" (what happens if, next to a sex shop, a space for children is built). Table 5 has the ranked candidates, identified as AX, for each strategy, and the resulting BordaCount ranking, including the score of each candidate. Table 6 has the answers of all the candidates in the previous table. This example also shows that not all strategies retrieve the same answers – e.g., some are introduced only when stemming or lemmatisation is performed (A6, A7), and others only when considering synonyms (A10, A11) – and combining different strategies broadens the search space.

Table 5 Top-5 candidate FAQs for the query "*o que acontece se perto de uma sexshop for construído um espaço para crianças*", with three search strategies plus the resulting ranking with BordaCount.

Strategy	**Top candidate FAQs**				
	1 (5 points)	**2 (4)**	**3 (3)**	**4 (2)**	**5 (1)**
VanillaQuestion	A1	A2	A3	A4	A5
StemQuestionAnswer	A6	A1	A7	A8	A9
LemmaSynsQuestionAnswer	A1	A10	A6	A7	A11
BordaCount	A1 (14 points)	A6 (8)	A7 (5)	A2 (4)	A10 (4)

Table 6 Content of answers used in the example of the previous table.

ID	**Content**
A1	*A sua instalação não impede o funcionamento da sex shop, ainda que sejam sujeitos a obras ou se verifique a alteração do respetivo titular.*
A2	*O Turismo de Portugal I.P. fixa um prazo não inferior a 30 dias para que o estabelecimento inicie o processo de autorização de utilização para fins turísticos.*
A3	*A instalação de estabelecimentos de alojamento local em edifícios construídos de raiz para o efeito não está impedida por lei...*
A4	*Estes estabelecimentos podem continuar a utilizar essa denominação mas dispõem de um prazo de cinco anos ...*
A5	*Não existe um uso de "alojamento local"...*
A6	*Tratando-se de um imóvel construído antes da entrada em vigor do Decreto-Lei n.º 38382 de 7 de agosto de 1951 ...*
A7	*Os estabelecimentos sex shop devem cumprir os seguintes requisitos: ...*
A8	*Estes estabelecimentos podem continuar a utilizar essa denominação mas dispõem de um prazo de cinco anos ...*
A9	*A instalação de estabelecimentos de alojamento local em edifícios construídos de raiz para o efeito não está impedida por lei ...*
A10	*Só pode registar o apartamento como estabelecimento de alojamento local se ...*
A11	*Não, mantendo-se o mesmo estabelecimento de alojamento local, apenas é necessário efetuar uma alteração ao respetivo registo, ...*

Table 7 has the results obtained by combining all the search strategies, excluding (Normal), including (Normal+Fuzzy) or exclusively with fuzzy versions (Fuzzy), plus a selection of three somehow complementary strategies (Three): VanillaQuestion, StemQuestionAnswer and LemmaSynsQuestionAnswer. BordaCount considered always the top-5 candidates.

The obtained results confirm that it might be wise to combine results of different strategies, as they often complement each other. In this case, when combining all the results, 15 (> 3%) more questions were answered correctly, when compared to the best search strategy.

Yet, some search strategies are very similar, so it might not always be necessary to consider them all. Among other combinations of three strategies tested, the one presented got the highest accuracy. Considering both fuzzy and normal searches, it answered one more question correctly than combining all the search strategies. This suggests that combining too many strategies, some of which similar, may result in bias and have a negative impact on the final results.

Table 7 Results of the BordaCount using different search strategies.

Combination	Correct answers	
All (Normal)	307	(68%)
All (Fuzzy)	313	(67%)
All (Normal+Fuzzy)	328	(73%)
Three (Normal)	317	(70%)
Three (Fuzzy)	313	(70%)
Three (Normal+Fuzzy)	329	(73%)

5 Conclusion and Future Work

We described how the IR library Lucene can be used in the development of a QA agent that finds answers to questions in Portuguese. This is often enough and avoids the development of an agent with dialog capabilities, possibly resorting to closed platforms that only provide limited control to the developer.

As others did for similar purposes [11, 13, 14, 5], we took advantage of Lucene's indexing and search capabilities, but also exploited other utilities offered by this library and tuned them to our specific context. This included the use of a synonym map, considering fuzzy searches, or adding search fields with the result of additional pre-processing, namely lemmatisation. Configurations based on the previous lead to the development different search strategies.

Adapting the agent to a domain is a matter of changing the underlying list of FAQs it should be able to answer. In order to test the agent, we used it to answer FAQs related to economic activity in Portugal. In this scenario, we also created an evaluation dataset with variations of the original questions, which enabled the comparison of different search strategies.

Results suggest that fuzzy searches may be an alternative to language-specific normalisation; that searching both in the question and in the answer is more fruitful; and that synonyms contribute to better results when the text is lemmatised. Yet, perhaps the most relevant finding is that the best results are obtained when different search strategies are combined with a voting method. This happens because some strategies end up being complementary.

The code of the developed QA agent, to be used with any list of FAQs or underlying agents, is available from `https://github.com/hgoliv/qa_agent`.

Despite the previous insights, results obtained also confirm that answering natural language questions is a challenging task and there is much room for improvement. Towards better results, several developments and experiments were left to perform. Some of them will be tackled in the future. For instance, we aim to analyse the impact of considering other related words, such as hypernyms, or using more synonyms, though possibly less reliable; the impact of choosing a similarity measure other than Lucene's default BM25; or even to compute accuracy considering not only the first search result, but also how close the correct answer is to the first (i.e., whether it is on the top-3 or top-5).

We also aim to test Semantic Textual Similarity measures for matching user input with FAQs, possibly integrating the results of our previous work [1]. This may be considered as an alternative to Lucene search mechanisms, or, to avoid performance issues, it can be applied to an initial selection of candidates by Lucene, as others did [5].

On the technical level, we aim to investigate how to integrate lemmatisation as a filter in a Lucene's Analyzer class, which would avoid the creation of additional fields for lemmatised versions of the text. As the lemma depends on the part-of-speech and the latter on the sequence of words, this would have to be done before stopword removal.

In order to test future configurations, a larger dataset will soon be created, with more FAQs and also more variations, created and/or validated by a larger crowd. Automatising the creation of such variations may resort both to search logs or paraphrase generation techniques [2].

A longer-term goal would be to consider both the session context and the user feedback in the selection of answers. The former should include the previous questions made and the latter may be used, for instance, for adjusting the weights of a voting method according to its reliability.

References

1 Ana Alves, Hugo Gonçalo Oliveira, Ricardo Rodrigues, and Rui Encarnação. ASAPP 2.0: Advancing the State-of-the-Art of Semantic Textual Similarity for Portuguese. In *Proceedings of 7th Symposium on Languages, Applications and Technologies (SLATE 2018)*, volume 62 of *OASIcs*, pages 12:1–12:17, Dagstuhl, Germany, June 2018. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

2 Anabela Barreiro and Luís Miguel Cabral. ReEscreve: a translator-friendly multi-purpose paraphrasing software tool. In *Proceedings of the Workshop Beyond Translation Memories: New Tools for Translators*, 2009.

3 Daniel Braun, Adrian Hernandez-Mendez, Florian Matthes, and Manfred Langen. Evaluating natural language understanding services for conversational question answering systems. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 174–185, 2017.

4 Annalina Caputo, Marco Degemmis, Pasquale Lops, Francesco Lovecchio, and Vito Manzari. Overview of the EVALITA 2016 Question Answering for Frequently Asked Questions (QA4FAQ) Task. In *Proceedings of Third Italian Conference on Computational Linguistics (CLiC-it 2016) & Fifth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2016)*, volume 1749 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2016.

5 Lei Cui, Shaohan Huang, Furu Wei, Chuanqi Tan, Chaoqun Duan, and Ming Zhou. SuperAgent: A Customer Service Chatbot for E-commerce Websites. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, System Demonstrations*, pages 97–102. ACL Press, 2017.

6 Peter Emerson. The original Borda count and partial voting. *Social Choice and Welfare*, 40(2):353–358, February 2013.

7 Erick R. Fonseca, Simone Magnolini, Anna Feltracco, Mohammed R. H. Qwaider, and Bernardo Magnini. Tweaking Word Embeddings for FAQ Ranking. In *Fifth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian*, volume 1749. CEUR-WS.org, 2016.

8 Hugo Gonçalo Oliveira. A Survey on Portuguese Lexical Knowledge Bases: Contents, Comparison and Combination. *Information*, 9(2), 2018.

9 Lynette Hirschman and Robert Gaizauskas. Natural Language Question Answering: the View from Here. *Natural Language Engineering*, 7(4):275–300, 2001.

10 Zongcheng Ji, Zhengdong Lu, and Hang Li. An Information Retrieval Approach to Short Text Conversation. *CoRR*, abs/1408.6988, 2014.

11 Valentin Jijkoun and Maarten de Rijke. Retrieving Answers from Frequently Asked Questions Pages on the Web. In *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, CIKM '05, pages 76–83, New York, NY, USA, 2005. ACM.

12 Oleksandr Kolomiyets and Marie-Francine Moens. A Survey on Question Answering Technology from an Information Retrieval Perspective. *Information Sciences*, 181(24):5412–5434, December 2011.

13 Govind Kothari, Sumit Negi, Tanveer A. Faruquie, Venkatesan T. Chakaravarthy, and L. Venkata Subramaniam. SMS Based Interface for FAQ Retrieval. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2*, ACL '09, pages 852–860, Stroudsburg, PA, USA, 2009. ACL Press.

14 Daniel Magarreiro, Luísa Coheur, and Francisco S. Melour. Using subtitles to deal with Out-of-Domain interactions. In *Proceedings of 18th Workshop on the Semantics and Pragmatics of Dialogue (SemDial)*, pages 98–106, 2014.

15 Arianna Pipitone, Giuseppe Tirone, and Roberto Pirrone. ChiLab4It system in the QA4FAQ competition. In *Fifth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian*, volume 1749. CEUR-WS.org, 2016.

16 Fabio Rinaldi, James Dowdall, Michael Hess, Diego Mollá, Rolf Schwitter, and Kaarel Kaljurand. Knowledge-Based Question Answering. In *Proceedings of the 7th International Conference on Knowledge-Based Intelligent Information and Engineering Systems (KES 2003)*, pages 785–792, Oxford, UK, September 2003. Springer-Verlag.

17 Stephen Robertson and Hugo Zaragoza. The Probabilistic Relevance Framework: BM25 and Beyond. *Found. Trends Inf. Retr.*, 3(4):333–389, April 2009.

18 Ricardo Rodrigues, Hugo Gonçalo Oliveira, and Paulo Gomes. NLPPort: A Pipeline for Portuguese NLP (Short Paper). In *7th Symposium on Languages, Applications and Technologies (SLATE 2018)*, volume 62 of *OASIcs*, pages 18:1–18:9, Dagstuhl, Germany, 2018. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

19 Yiping Song, Cheng-Te Li, Jian-Yun Nie, Ming Zhang, Dongyan Zhao, and Rui Yan. An Ensemble of Retrieval-Based and Generation-Based Human-Computer Conversation Systems. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pages 4382–4388. International Joint Conferences on Artificial Intelligence Organization, July 2018.

20 Oriol Vinyals and Quoc V. Le. A Neural Conversational Model. In *Proceedings of ICML 2015 Deep Learning Workshop*, Lille, France, 2015.

21 Ellen M. Voorhees. The TREC Question Answering Track. *Nat. Lang. Eng.*, 7(4):361–378, December 2001.

22 Joseph Weizenbaum. ELIZA: a computer program for the study of natural language communication between man and machine. *Commun. ACM*, 9(1):36–45, January 1966.